Advances in Performance Analysis, Vol. 1 (1), 1998, pp. 1-37

Performance of MAC Protocols for Broadband HFC and Wireless Access Networks*

Kotikalapudi Sriram
Bell Labs - Lucent Technologies
101 Crawfords Corner Road
Holmdel, NJ 07733-3030
sriram@lucent.com

Abstract: We present a brief overview of some key features that enable superior performance and efficiency in Medium Access Control (MAC) protocols for Hybrid Fiber-Coax (HFC) and wireless access networks. These features include: (1) minislots for bandwidth allocation granularity, (2) flexible transport capability for fixed-length (ATM) protocol data units (PDUs) and variable-length (VL) PDUs (e.g., IP, 802.3 frames, IPv6), (3) means for low delay transport of STM voice information, (4) dynamically variable minislots for contention request resolution, and (5) piggy-backing for additional transmission requests. We also include in this study a comparison of some alternative MAC protocol features. The objective of the study is to identify a set of MAC protocol features that results in significant performance and capacity gain, and (in conjunction with connection level bandwidth management algorithms) is capable of meeting different delay and bandwidth requirements for different traffic classes such as telephony, data, video and multi-media. The resulting MAC protocol would enable efficient integration of various applications in a community access network employing HFC or wireless media. The IEEE 802.14 standards specification (draft) and the MCNS (a cable industry consortium) specification on cable modems have embraced several of the key features discussed here as part of their MAC protocols.

Key Words: Broadband Access, Hybrid Fiber-Coax, Wireless Networks, Medium Access Control, Performance Modeling and Analysis, Asynchronous Transfer Mode, Internet Protocol, Wireless ATM, LAN.

1. INTRODUCTION

Experiences over the last several years and projections into the future suggest an incipient revolution in the variety of communication services offered to the user, and in the infrastructure used to support these services. Internet access, client applications with remote servers, video-on-demand, and interactive multi-media are among the new services that will supplement traditional telephony. Concomitantly, Frame Relay, ATM and SONET are enlarging the capabilities of networks at all geographical levels. Wireless and cable-based access technologies are providing attractive alternatives to wireline loop and analog modem-based access. Collectively, these promise to change the means and price of future communication services.

The new services are characterized by a plethora of throughput rates and delay requirements, along with varying degrees of burstiness. Looking to the future, combinations of these transmission attributes will become even richer and will create new technical challenges for equipment designers and network operators. An important goal of the system architecture and protocol design for hybrid fiber-coax (HFC) and wireless networks is service integration and real time statistical multiplexing of various traffic types. This will allow efficient use of scarce upstream (i.e., from home/mobile to network) bandwidth, amortization of equipment (switches, routers, operations systems, RF modems, etc.) and plant upgrade costs across multiple services.

Currently, work is in progress to design Medium Access Control (MAC) protocols which enable efficient utilization of bandwidth in upstream and downstream channels over broadband access media, e.g., HFC and wireless [1]-[16]. The upstream and downstream transmissions take place in separate RF channels occupying different parts of the RF spectrum. The available total bandwidth is usually much more in the downstream direction than in the upstream direction. An HFC network typically has several dozens of RF channels each at a rate of about 10 Mb/s to 40 Mb/s in the downstream, and a dozen or so RF channels each at a rate of 2.56 Mb/s in the upstream [2][3]. The downstream transmission takes place in a broadcast mode with each MAC protocol data unit (PDU) carrying the address of a station to which it is destined. The upstream bandwidth is shared by many stations using a multiple access protocol, operating under the control of a central controller which may be located in the head-end of an HFC network or the base station of a wireless network. In the rest of this paper, we use HFC terminology for ease of communication (with an understanding that the corresponding wireless terminology can be substituted when the MAC protocol is discussed in the context of wireless networks). In the context of HFC networks, the central controller is referred to as a Head-end Controller (HC). The stations are referred to as Cable Modems (CM).

Figure 1-1 shows how a MAC protocol fits into an overall systems architecture for HFC networks supporting a multi-service environment including voice, data, video, and multi-media applications. The architecture illustrated in Figure 1-1 calls for a flexible MAC protocol capable of transporting information in Synchronous Transfer Mode (STM), fixed-size Asynchronous Transfer Mode (ATM) Protocol Data Units (PDUs), or Variable Length (VL) PDUs (e.g., IP, 802.3 packets). Such a MAC protocol would also be bandwidth efficient, taking advantage of statistical multiplexing of bursty traffic and providing low latency for delay sensitive traffic. ATM and VL PDU sizes are integral multiples of a basic upstream unit such as a minislot which will be discussed later. When VL PDUs are used for transport, the information is carried in a format close to its native format (e.g., IP, 802.3, MPEG-2, etc.) with minimum additional overhead in the access network. An early deployment may support only a subset of the functions shown in Figure 1-1. An example of possible early deployment is to support data services over IP and 802.3 using VL PDUs. The backbone connectivity is through IP routers, while the connectivity between the CM and Information Appliances (IAs) is over 802.3 LAN (a true LAN or a point-to-point connection using LAN protocols). Another early deployment scenario may involve video retrieval with MPEG-2 packets carried as native data with tighter jitter requirements or over ATM cells with an appropriate service class. MAC protocols are being designed to allow a smooth migration from an early deployment of limited set of services to a full service access network with connectivity to multiple backbones and multiple services [1]-[9][14]-[16].

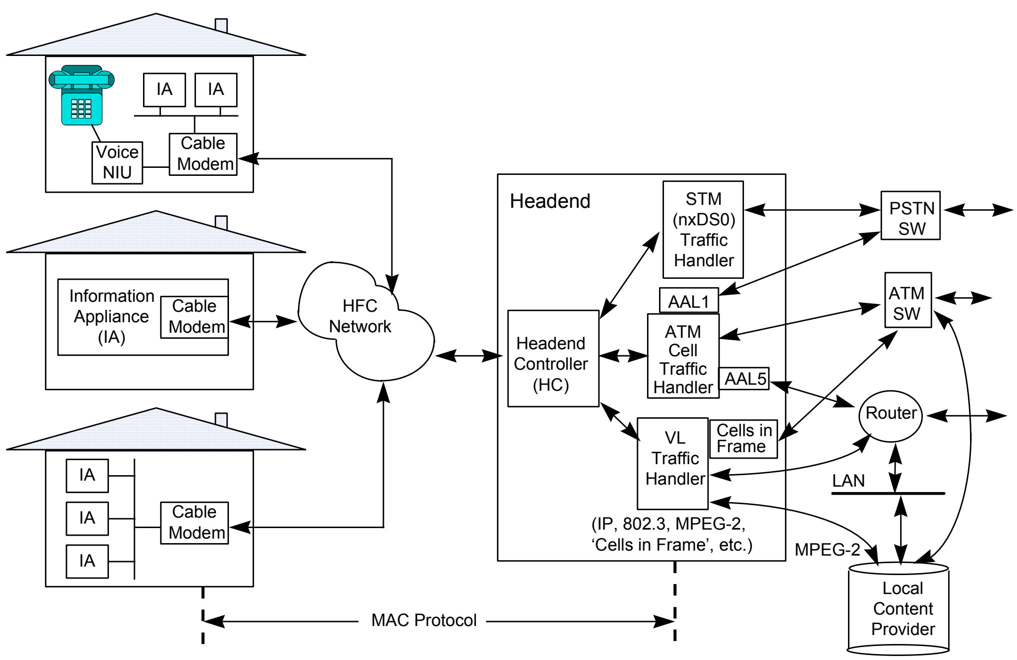

Figure 1-1: The role of a MAC protocol in HFC networks.

In this paper, we present a brief overview of some key features that enable superior performance and efficiency in Medium Access Control (MAC) protocols for Hybrid Fiber-Coax (HFC) and wireless access networks. These features include: (1) minislots for bandwidth allocation granularity, (2) flexible transport capability for fixed-length PDUs (ATM cells) and variable-length (VL) PDUs (e.g., IP, 802.3 frames, IPv6), (3) means for low delay transport of STM voice information, (4) dynamically variable minislots for contention request resolution, and (5) piggy-backing for additional transmission requests. We also include in this study a comparison of some alternative MAC protocol features. The objective of the study is to identify a set of MAC protocol features that results in significant performance and capacity gain, and (in conjunction with connection level bandwidth management algorithms) is capable of meeting different delay and bandwidth requirements for different traffic classes such as telephony, data, video and multi-media. The resulting MAC protocol would enable efficient integration of various applications in a community access network employing HFC or wireless media. The IEEE 802.14 standards specification (draft) and the MCNS (a cable industry consortium) specification on cable modems have embraced several of the key features discussed here as part of their MAC protocols [14]-[16].

2. PRINCIPLES OF MAC PROTOCOL OPERATION

We consider here MAC protocols in which a head-end controller (HC) manages the upstream and downstream components of the MAC protocol. The protocol operates in one-to-many broadcast mode in the downstream, and many-to-one multiple access mode in the upstream. The HC and cable modems (CMs) have a common notion of upstream and downstream frame structures based on initial synchronization and periodic control messages from HC to CMs regarding possible restructuring of the upstream and downstream frames. Because the upstream bandwidth is scarce compared to the downstream bandwidth, multiple upstream RF channels are usually associated with one downstream RF channel [2][3].

2.1. *Upstream Frame Structure and Protocol Operation*

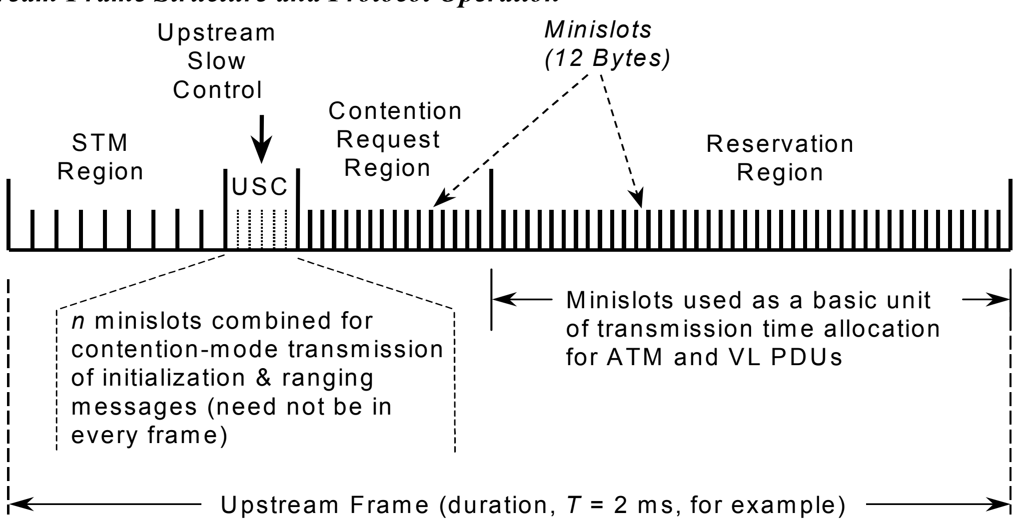

Figure 2-1: An example upstream frame structure showing STM slots, contention request minislots, and a PDU reservation region with minislot as a basic unit of granularity.

In the upstream component of the MAC protocols, minislots are used to request permission to transmit packets of information (voice, data, video, or multi-media) in the upstream channels, and the information is subsequently transmitted in packet time-slots allocated by a central controller [3][4]. The requests transmitted in request minislots are subject to collisions, and an appropriate Collision Resolution Algorithm (CRA) is used to resolve collisions. The CRA specifies the rules by which the CMs involved in a collision can retransmit to resolve the collision. A brief overview of different types of CRA is given in

Section 2.2 (also see [4]). The HC transmits status (feedback) messages in a downstream channel to inform the CMs about the collision status (collision or success) of requests in minislots. For the requests received successfully, the status message also includes grants of upstream packet slots in which the actual information may be transmitted.

An example upstream frame structure is shown in Figure 2-1. It includes an STM region, a contention request region, and a reservation region for ATM and Variable Length (VL) PDUs. Synchronous traffic such as DS0 and n*DS0 calls are allocated fixed size time slots in the STM region. The frame duration is denoted by T ms. The frame duration, T, in Figure 2-1 is shown to be 2 ms as an example. The round-trip propagation delay accounts for about 1 ms, and about 1 ms is budgeted for transmission and processing times at HC and CM. Each time slot in the STM region contains the information bytes for one STM connection collected over T ms plus physical layer (PHY) and MAC overheads. These overheads include guard time, preamble, and forward error correction (FEC). The contention region consists of multiple minislots. As mentioned before, requests for transmission are made in the contention request minislots. When the request successfully reaches the HC, a reservation is granted to the requesting station for transmission of its PDU in the reservation region of a future frame. The reservation region (see Figure 2-1) also consists of minislots of the same size as the request minislots. The reservation for PDU transmission is allocated in units of minislots. The size of a minislot is selected such that an ATM PDU burst equals an integral number of minislots. A variable number of minislots are allocated for VL PDUs depending on the VL payload size.

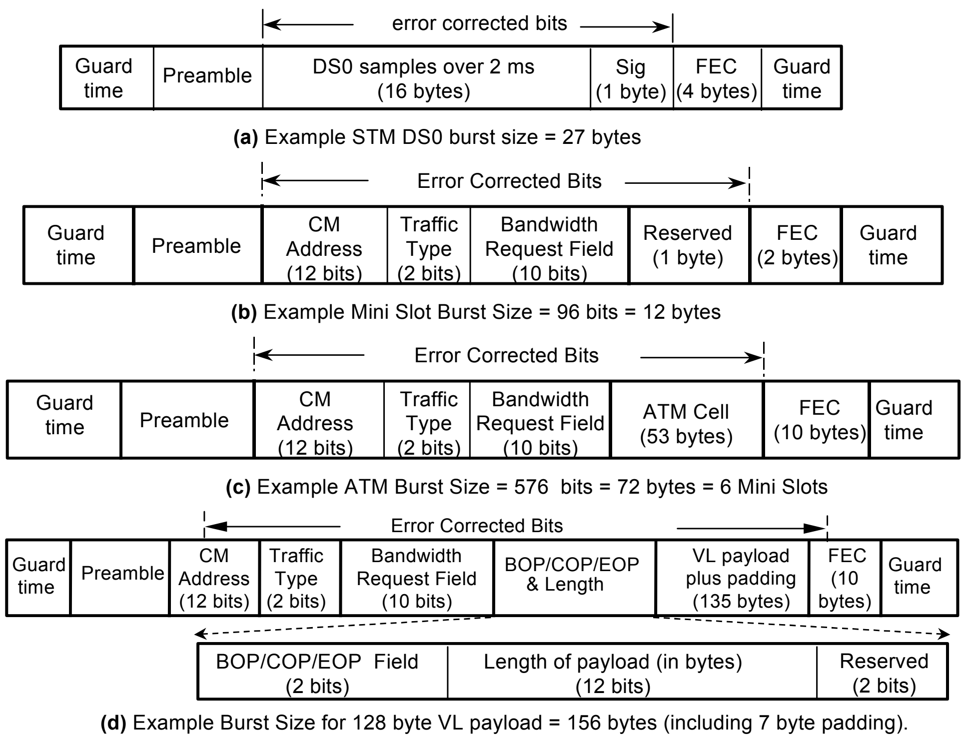

Figure 2-2: Example formats of upstream bursts for: (a) STM, (b) minislot request, (c) ATM PDU, and (d) VL PDU.

Upstream bursts of different types are illustrated in Figure 2-2 based on the example of Adaptive Digital Access Protocol (ADAPt+) described in [3]. In this example, it is assumed that QPSK modulation is used in the upstream at 1.28 M symbols/s rate in approximately 1.8 MHz RF channels resulting in 2.56 Mb/s data rate per RF channel. The guard time and preamble are of the same size for all types of bursts. The size of the FEC is a function of the payload size, and is different for different size bursts. The payload of the STM burst in Figure 2-2 is 16 bytes corresponding to frame (packetization) duration, T = 2 ms. The Sig field in the STM burst carries A-H signaling bits of the DS0 signaling scheme. The size of the STM burst is 27 bytes in this example. The minislot request burst in Figure 2-2 is of size 12 bytes. The cable modem address is shown to be 12 bits long. The traffic type field specifies if the request is for an ATM or VL PDU. The bandwidth request field specifies the length of the PDU for which request is being made. The unit of bandwidth request is an ATM cell when the traffic type is ATM, and the unit is a minislot when the traffic type is VL. An ATM burst (see Figure 2-2) is 72 bytes long including the PHY/MAC overheads, and is equal to six minislots. The VL PDU exemplified in Figure 2-2 equals 13 minislots in length. It carries a VL payload of size 128 bytes (with 7 bytes padded to make the total burst length an integer multiple of minislots. The 2-bit BOP/COP/EOP field in the VL PDU is to indicate beginning (00), continuation(01), and end (10) of packet in case of segmentation of long VL payloads into smaller MAC layer VL PDUs. When the 2-bit field is set to a value 11, it indicates to the higher layer VL data element that no segmentation was done. For several additional details related to upstream frame and PDU structures, the reader is referred to [3][14][16].

Piggy-backing of Requests:

A request made for an additional PDU transmission time slot, while transmitting a presently allocated PDU burst, is called a piggy-backed request. In Figure 2-2, the ATM PDU and the VL PDU bursts include piggybacked requests. The piggy-backed request field is identical to that in a request minislot. The advantage of piggy-backing is that a request for an additional PDU waiting in queue (if any) can be made by a CM without going through further contention. Thus, piggy-backing by CMs that currently have PDU allocations helps reduce the traffic and probability of collisions in contention minislots, and thereby benefits other CMs which are transmitting requests in a contention mode.

Propagation Delay Effects and Upstream Frame Duration:

The status message is subject to a round-trip propagation delay due to the physical distances between the HC and the CMs. The round-trip propagation delay together with the HC and CM processing delays for processing and notification of collision status information give rise to an upstream frame structure in at least two possible ways (see Figure 2-3 and Figure 2-4). In Figure 2-3, the minislots are all clustered adjacently at the beginning of each upstream frame, while in Figure 2-4, they are distributed over the upstream frame interval. The frame duration, T_F , for the two cases is lower-bounded as follows:

where t_R is the total time duration of the request minislots placed at the beginning of a frame; $2*t_p$ is the round trip propagation delay; t_{HC} and t_{CM} are the Head-end Controller (HC) and Cable Modem (CM)

$$T_F \geq t_R + 2\ t_p + t_{HC} + t_{CM} \quad \text{(for mini-slots clustered at beginning of frame)},$$

$$T_F \geq 2\ t_p + t_{HC} + t_{CM} \quad \text{(for mini-slots distributed over the frame interval)},$$

processing times. The upstream frame structure can be chosen to be either of the two illustrated above. For the case of MAC protocol operation as per Figure 2-3, the status message corresponding to a cluster of minislots is received prior to the beginning of the next frame. For the case of MAC protocol operation as per Figure 2-4, the status message corresponding to minislots in a frame is received during the next frame. The second case (Figure 2-4) allows the flexibility of placing minislots anywhere in the frame, and hence it would require less frequent regrouping of idle time slots in a frame. It also requires two instances (or engines) of the Contention Resolution Algorithm (CRA) to be run in parallel (in an interleaved manner) - one over odd numbered frames and another over even numbered frames. However, this requirement is not expected to add any significant implementation complexity. The protocol features and results of performance analysis presented in this paper are equally applicable to either of the two cases of upstream MAC operation depicted in Figure 2-3 and Figure 2-4.

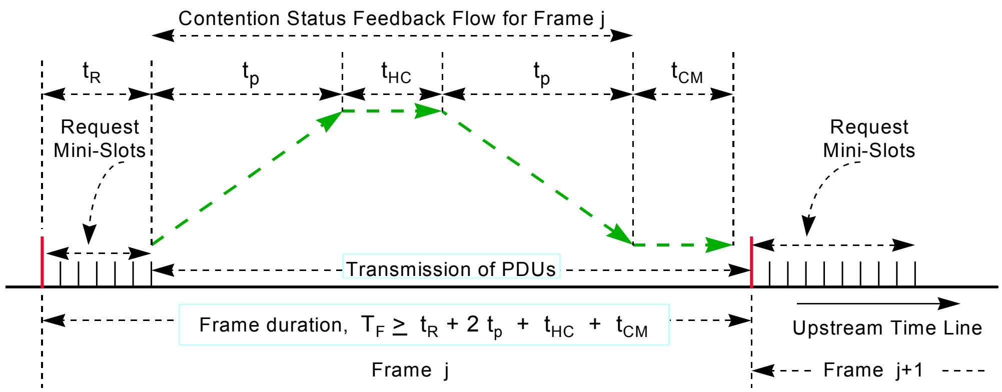

Figure 2-3: Upstream frame structure with contention request minislots grouped together at the beginning of each framing interval.

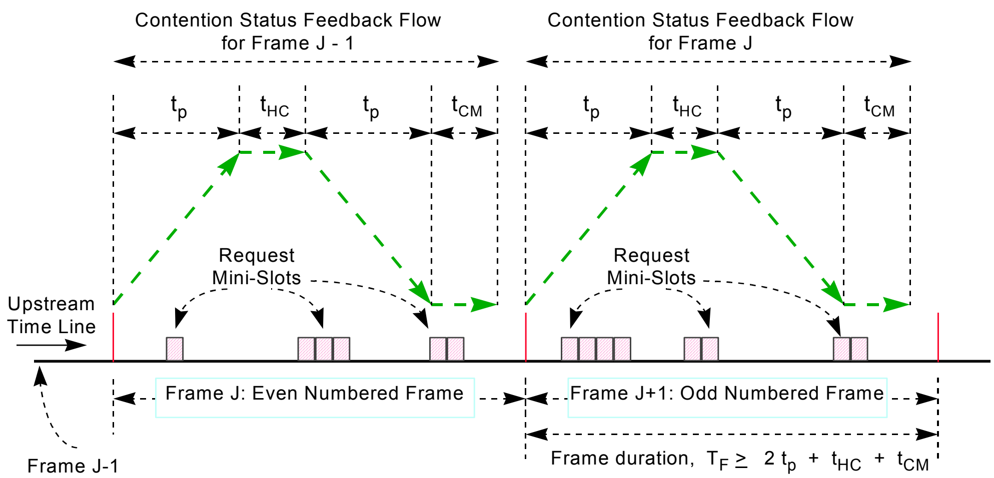

Figure 2-4: Upstream frame structure with contention request minislots distributed over the frame interval.

2.2. *Principles of Collision Resolution Algorithms*

A collision occurs in a minislot of an upstream frame when two or more Cable Modems (CM) transmit request messages in the same minislot. Collisions are detected at the Head-End Controller (HC), and status messages are transmitted back to the CMs. A Collision Resolution Algorithm (CRA) is employed to resolve collisions. CRA can be broadly classified into three categories as briefly described below. Although we assume the use of minislots, the descriptions of CRA provided here are valid generally in the context of any contention-oriented packet transmission.

Random Binary Exponential Back-Off Algorithm:

The cable modem generates a random number, i, uniformly distributed between $[0, (2^j - 1)]$, where j is the number of collisions that the CM has experienced for the request message it is attempting to transmit. If j is larger than 10, then i is selected from a uniform distribution in the range $[0, (2^{10} - 1)]$. The cable modem skips i minislot transmission opportunities, and retransmits the request in the $(i + 1)^{th}$ minislot (these minislots may span over one or more upstream frames).

Adaptive p-Persistence Algorithm:

In the adaptive p-persistence algorithm, a parameter, say q, is communicated from the HC to the CMs as part of the status information. Upon receiving the status information, each CM generates a uniform random number, i, such that $1 \leq i \leq q$. If $i \leq k$, where k is the number of minislots available for request transmissions in the forthcoming frame, the CM transmits its request in the i^{th} minislot, else it waits for the next status message and repeats the operation by generating a new random number. The HC can vary the value of q from one frame to the next as a function of the number of collisions that are observed in a frame [9][13]. The value of q may be increased or decreased when the number of collisions observed at the HC increases or decreases, respectively. In effect, the algorithm adaptively decreases or increases the probability of retransmission by a CM as a function of the existing traffic load in the system. When the traffic load is heavier resulting in more frequent collisions, then the value of q is increased. When the traffic load is lighter resulting in less frequent collisions, then the value of q is decreased.

Tree Algorithm:

A tree algorithm may use a binary, ternary, or n-ary tree to help spread re-transmissions and thereby resolve collisions at each consecutive step in the contention resolution process. Although binary tree is commonly used, it has been shown that the ternary tree algorithm provides higher performance [10][11]. We briefly describe here the operation of a ternary tree algorithm.

The principle of operation of a ternary tree CRA is as follows (see [10]-[12] for more details). Clock time increments from left to right within an upstream frame (as depicted in Figure 2-3 and Figure 2-4). Each CM listens to the HC status message for the preceding frame, and determines therefrom the number of minislots, say k, with collisions that were located to the left of the minislot in which it transmitted its request (unsuccessfully). The CM skips the next $3*k$ minislots (which may be contained in one or more forthcoming frames), and then it retransmits its request by randomly selecting one of next three minislots. While one group of transmission requests are being resolved for collisions, new requests are held in buffers at the CMs, and transmitted only after the current group has been resolved completely. From one frame to the next, the span of minislots over which retransmissions occur spreads wider by three times the number of minislots with collisions. Therefore, over successive frames, the probability of further collisions rapidly decreases within a group of contention requests that are being resolved. When this progressive process of contention resolution finally results in a frame with one or more successful request messages and no collisions in any of the minislots, then the next frame is opened for the waiting new requests to be transmitted. A fresh set of collisions may occur; the operation of the CRA as described before repeats, and a new cycle of collision resolutions occurs.

The first two CRA algorithms described above have the same upper bound on the throughput per minislot which is the same as that for slotted Aloha (about 37%). The corresponding throughput of a ternary tree CRA can be somewhat higher (up to 50%). The ternary tree and adaptive p-persistence are deemed more desirable for minimizing delay variance and maximizing stability of the system from a traffic view point. A performance comparison in terms of delay, delay variance, stability, etc. between ternary tree and adaptive p-persistence algorithms is provided in [9]. The IEEE 802.14 specification (draft) [14] includes a ternary tree CRA as part of its MAC protocol. The MCNS (a cable industry consortium) document [15][16] specifies the use of a random binary exponential back-off algorithm as part of its MAC protocol requirement. Because different implementations may use different CRAs, for the purpose of an analytical study of different possible ways of use of minislots, we conservatively assume that the throughput efficiency of minislots is 37% (which is achievable by use of any of the CRAs described above).

2.3. *Down Stream Frame Structure and Protocol Operation*

Each downstream RF channel (e.g., 6 MHz wide) is suitably modulated (e.g., using 64- or 256-QAM) to provide a digital bit stream (e.g., 30 or 40 Mb/s). Within each RF channel a frame structure such as shown in Figure 2-5 may be used to organize and schedule the transmission of STM, ATM, and VL information. For the purpose of supporting low delay STM, a 125 μs subframe may be used, and eight successive subframes would constitute a 2 ms downstream frame to match with a 2 ms upstream frame [3]. The framing bits (8 bits) help the CM to synchronize with the downstream frames. If telephony traffic were not supported in a strictly STM mode of transport in the downstream, then the downstream framing as shown in Figure 2-5 is not necessary. The downstream synchronization may be derived by Header Error Check (HEC) sequence in the ATM and VL PDU headers. Additional details with regard to frame organization and upstream/downstream synchronization are provided in [3][14][16].

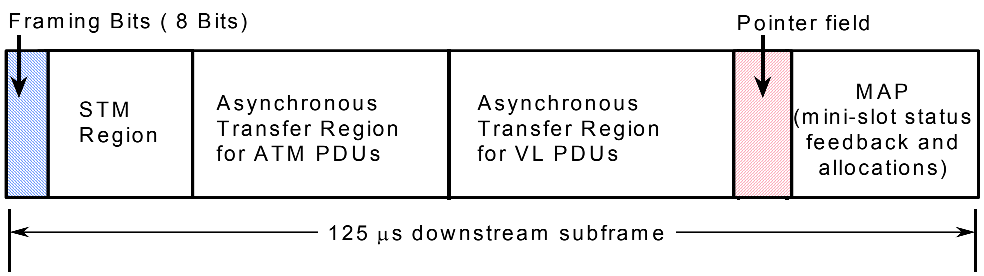

Figure 2-5: An example downstream frame structure.

Example formats of ATM and VL PDUs are shown in Figure 2-6 and Figure 2-7. The CM address provided in the ATM and VL PDU headers is used by CMs to pick out and process the PDUs that are intended for them. The asynchronous transfer region for ATM and VL PDUs may be separated into two byte-streams within the larger pipe (of 30 Mb/s) with a boundary as shown in Figure 2-5. This arrangement facilitates easier quality of service (QoS) guarantees to ATM connections without interference due to possibly long VL PDUs. On the other hand, the ATM and VL PDUs may be transmitted within a common asynchronous region or byte-stream provided that the longer VL PDUs are segmented. The purpose of segmenting the longer VL PDUs would be for avoiding excessive delay jitter for ATM PDUs which would in turn help meet ATM QoS requirements. The pointer field (PF) in Figure 2-5 points to the beginning of the first full VL PDU in the next subframe. This helps as an additional means for synchronization in the downstream within the VL PDU region.

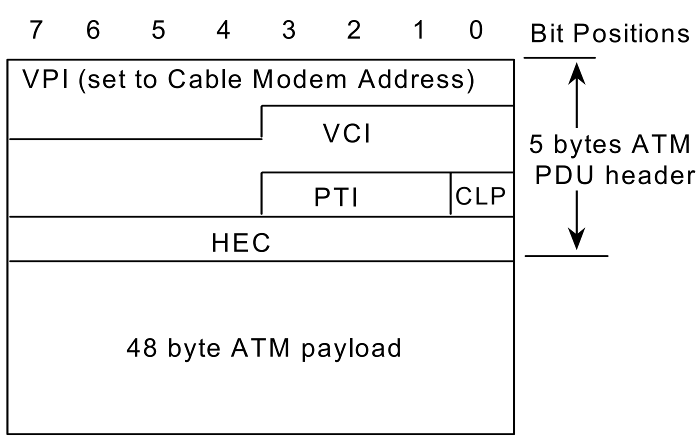

Figure 2-6: An example downstream ATM PDU format.

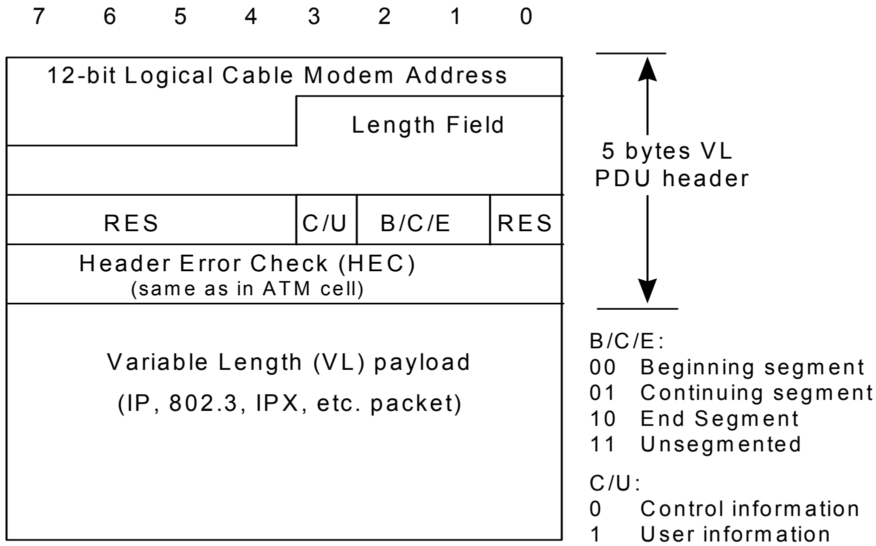

Figure 2-7: An example downstream VL PDU format.

The MAP field in Figure 2-5 provides a map of upstream minislots including collision status in the prior upstream frame, the number and locations of contention request minislots in the next upstream frame, and information about allocation of upstream minislots to CMs for ATM and VL PDU transmissions.

Low-delay requiring synchronous DS0 (e.g., telephony) connections may be carried in one of three possible ways:

(a) Using an STM byte region in 125 μs subframes:

One byte per DS0 is transmitted in an STM region within 125 μs subframes (as shown in Figure 2-5). This arrangement results in the lowest possible overhead for STM traffic while providing strictly synchronous DS0 (or n*DS0) service capability. One signaling byte overhead (per 125 μs) is allowed to accommodate the A-H signaling bits for up to 16 DS0 connections.

(b) Scheduling composite ATM cell every 125 μs:

One or more ATM cells are scheduled at 125 μs intervals (approximately), and each DS0 connection is allocated one byte in an ATM cell payload. Roughly one signaling byte overhead is required for each group of 16 DS0 bytes to accommodate the A-H signaling bits. Hence, three bytes per ATM cell payload are allowed for signaling. Each DS0 connection is allocated a byte in a fixed position within an ATM cell payload of 48 bytes. Thus, up to 45 DS0 connections can be position multiplexed in each ATM cell (scheduled at 125 μs intervals). ATM cell scheduling can be done such that the delay jitter does not exceed 125 μs and the downstream STM delay is bounded to 250 μs.

(c) Scheduling composite ATM cell every 1 ms:

One or more ATM cells are scheduled at 1 ms intervals (approximately), and each DS0 connection is allocated eight bytes in an ATM cell payload. Roughly, one signaling byte overhead is required for each group of 16 DS0 bytes to accommodate the A-H signaling bits. Hence, three bytes per ATM cell payload are allowed for signaling. Each DS0 connection is allocated eight bytes in a fixed position within an ATM cell payload of 48 bytes. Thus, up to 5 DS0 connections can be position multiplexed in each ATM cell (scheduled at 1 ms interval). ATM cell scheduling can be done such that the delay jitter does not exceed 125 μs and the downstream STM delay is bounded to 1.25 ms.

Section 3 describes data traffic characteristics and assumptions used in our analysis and numerical results. In Sections 4 and 5, we develop a performance model to compare message delays and bandwidth efficiencies for carrying native protocol data units (IP, 802.3 packets) over ATM vs. VL PDUs in the upstream and downstream channels, respectively. Section 6 describes, analyses and numerically compares MAC protocols with no minislots, with fixed number of minislots, and with dynamically variable number of minislots. In Section 7, we quantitatively compare the bandwidth efficiencies of the three different modes of carrying synchronous DS0 traffic described above.

3. TRAFFIC CHARACTERISTICS

For the purpose of this study, the traffic supported by the system is considered to be primarily of two types: (1) reservation-oriented long messages (e.g., file transfers, constant bit-rate (CBR) connections) that require a fixed bandwidth allocation for a relatively long period of time, and (2) random burst messages which arrive as batches (or bursts) of packets at random intervals. The former typically generate one contention message at the beginning of each transaction to request bandwidth for the entire duration of the transaction. The latter type of traffic sources generate frequent and random contention messages, one for each burst of data packets. In this paper, we refer to the former as reservation-oriented traffic and the latter as contention-oriented burst traffic.

The contention-oriented burst traffic is assumed to have different message lengths (in number of packets). For example, when Ethernet frames of sizes (64, 128, 256, 512, 1024, 1516) bytes are converted to fixed-size ATM cells using AAL-5 adaptation, the messages would be of lengths (2, 3, 6, 11, 22, 32) ATM cells, respectively. From available traffic measurement studies [17], it is known that the frequency of Ethernet frames are approximately (0.6, 0.06, 0.04, 0.02, 0.25, 0.03), for the respective sizes of (64, 128, 256, 512, 1024, 1516) bytes. The average message length in that case computes to 8.33 packets (ATM cells) per message. The message arrival process is assumed to be Poisson. Thus the contention-oriented burst traffic is characterized by a Poisson batch arrival process; the distribution of batch size (in number of packets) is derived based on the MAC PDU payload size information and the message length distribution stated above. The messages may be carried using ATM or VL PDUs. For ATM PDUs, the MAC PDU payload size is 48 bytes. For VL PDUs, the MAC PDU payload is equal to the message size whenever it is smaller than 256 bytes (the maximum PDU payload size permissible for VL PDUs is assumed to be 255 bytes based upstream FEC limitation). When the message length is larger than 255 bytes, it is segmented and transmitted in multiple VL PDUs.

4. UPSTREAM BANDWIDTH EFFICIENCY COMPARISON FOR ATM VS. VL PDU

In this section, we develop an analytical model for the upstream message delay. The goal is to compare the performance and throughput efficiency of using ATM PDU vs. VL PDU for message transmission over the upstream channel. The MAC protocol based on minislot granularity (described in Section 2.1) is flexible so that the messages (higher layer data units, e.g., IP, 802.3 packets) can be carried in either VL PDUs or in ATM PDUs. In the latter case, the messages are segmented using AAL-5. The queuing model developed here is generic and applicable to either case. The comparison between ATM and VL PDU usage is best highlighted if the traffic is assumed to be 100% random bursty messages (see Section 3), i.e., all sources (CMs) produce contention-oriented burst traffic. This would generally be true for the upstream traffic in internet applications.

The queuing model developed here is based on a combination of delay analysis of contention access for the request messages [2] and an $M/G/1$ approximation for message queuing [19][20] for transmission over an upstream channel. The upstream message access delay, W_U , is defined as the time from arrival of the first byte of a message at a cable modem (CM) to the time the last byte is received at the head-end controller (HC). This includes the initial contention request delay, the HC processing and queuing delay for issue of the grants, the round-trip propagation delay, and the message transmission delay. The HC is assumed to provide grants to the CMs on the basis of receiving successful requests in the minislots on a first-come-first-served (FIFO) basis. Since the CMs are assumed to produce data in the same manner (i.e., Poisson batch arrivals at a certain rate), the analysis given here also holds good for the round-robin service discipline [20][21].

W_U can be determined as follows by adding the various contributing components:

$$W_U = D_c + a_s * (Q_U + T_U) + \tau_p \tag{4-1}$$

where,

D_C = request contention access delay (ms),

a_S = average number of PDUs per message (computed from message size distribution and segmentation information),

Q_U = upstream queuing delay at PDU level due to HC scheduling process,

T_U = average PDU transmission time in the upstream, and

τ_p = one-way propagation time (ms).

The request contention access delay, D_C , is the delay for a request message to reach the HC successfully. We assume that a random binary exponential algorithm is used for collision resolutions. For a conservative approximation for W_U , we use a high percentile value of D_C , such as the mean plus five times standard deviation. We use the results presented in an earlier paper (see Section 8.1, Figure 13 in [2]) to obtain the following approximation for D_C :

$$D_c = 2 * F * \left(1 + \frac{3 * \rho}{2}\right) \tag{4-2}$$

where,

F = upstream frame duration (2 ms), and

ρ = upstream channel utilization due to all sources.

In deriving (4-2), we use data from Figure 13 in Section 8.1 of [2] corresponding to message size = 1, and knowledge that the x-axis there should be scaled up by at least three times to convert from throughput in minislots to data utilization, ρ . For a more detailed understanding of the relationship between minislots throughput and data message throughput, please see [4].

We have not included the effect of piggy-backing of requests in the model developed here. We assume that each message generates one request individually. Piggy-backing of requests would help improve the performance for ATM PDUs as well as VL PDUs by lowering the contention access delay component, D_C .

Let us define the following parameters for use in deriving an equation for Q_U:

R = source rate (kb/s),

P = average message size without overhead (bytes),

P_S = avgerage PDU size including overhead (bytes),

P_X = average PDU size without overhead (bytes),

$a_S = P/P_X$ = average number of PDUs per message,

λ = average message rate per source (messages/ms),

S_U = average PDU service time in the upstream,

ρ = upstream channel utilization per RF channel,

C = upstream aggregate channel bandwidth over all RF channels available for data (kb/s),

C_1 = upstream channel bandwidth per RF channel (kb/s),

C_m = upstream channel bandwidth used per RF channel for provisioning minislots (kb/s),

$g_m = C_m / C_1$ = fraction of upstream channel bandwidth used for provisioning minislots,

β = coefficient of variation of PDU service time (derived from PDU size distribution), and

σ_U = standard deviation of upstream message delay (ms).

The following equations hold:

$$\lambda = \frac{R}{8 * P} \quad \text{(packets per ms)} \tag{4-3}$$

$$S_U = \frac{8 * P_s}{C * (1 - g_m)} \quad \text{(ms)} \tag{4-4}$$

$$\rho = \lambda * S_U * N = \frac{R * P_s * N}{P * C * (1 - g_m)} \tag{4-5}$$

$$T_U = \frac{8 * P_s}{C_1 * (1 - g_m)} \quad \text{(ms)} \tag{4-6}$$

Now using an *M/G/1* queuing model at the PDU level (see [19][20]), we obtain the following set of equations for upstream performance measures of interest, i.e., Q_U, W_U , and σ_U :

$$Q_U = \frac{\rho * S_U * (1 + \beta^2)}{2 * (1 - \rho)} \quad \text{(ms)} \tag{4-7}$$

$$W_U = 2 * F * \left(1 + \frac{3 * \rho}{2}\right) + a_s * \left(\frac{\rho * S_U * (1 + \beta^2)}{2 * (1 - \rho)} + \frac{8 * P_s}{C_1 * (1 - g_m)}\right) + \tau_p \quad \text{(ms)} \tag{4-8}$$

$$\sigma_U \approx a_s * Q_U * \sqrt{\frac{4 - \rho}{3 * \rho}} \quad \text{(ms)} \tag{4-9}$$

4.1. ***Numerical Examples for Comparison of ATM vs. VL PDU in Upstream***

Messages belonging to random bursty traffic (described in Section 3) may originate from home computers with an interface such as a 10baseT (802.3) to the CMs. These messages can be transported using the VL PDUs or ATM PDUs (with AAL-5 adaptation in the latter case). When these packets are segmented into smaller ATM PDUs, the upstream bandwidth efficiency suffers because of (a) partially filled ATM cells, (b) ATM header overhead, and (c) repeated PHY & MAC layer overheads for each segment. The VL mode would also require segmentation for payloads over 256 bytes because of FEC consideration, but this level of segmentation is much less compared to the segmentation required by the ATM mode. For these reasons the overheads for carrying the messages in upstream channels is much lower for VL PDUs as compared to that for the ATM PDUs (see Figure 4-1 and Table 1). The upstream bandwidth efficiencies for ATM and VL PDUs as a function of message sizes are compared in Figure 4-2. The average bandwidth efficiencies for ATM and VL PDUs (averaged over the packet size distribution of Section 3) are also shown in Figure 4-2; see the rightmost pair of bars.

The upstream message access delay is defined as the time from arrival of the first byte of a message (or packet) at a cable modem (CM) to the time the last byte is received at the head-end controller (HC). This includes the initial contention request delay, the HC processing and queuing delay for issue of the grants, the round-trip propagation delay, and the packet transmission delay. The delay plotted in Figure 4-3 is the mean plus five times the standard deviation of message access delay. The total upstream bandwidth is assumed to be 9 Mb/s (three RF channels each providing 3 Mb/s bandwidth). Clearly, VL PDUs facilitate lower delays as well as higher throughput. If the delay requirement were 20 ms, then from Figure 4-3 it can seen that VL PDUs allow a per user throughput of 120 kb/s whereas ATM PDUs would allow only 80 kb/s (for 50 users sharing 9 Mb/s). For the upstream channels, VL PDUs provide about 50% higher throughput and capacity over ATM PDUs for the random bursty traffic considered here.

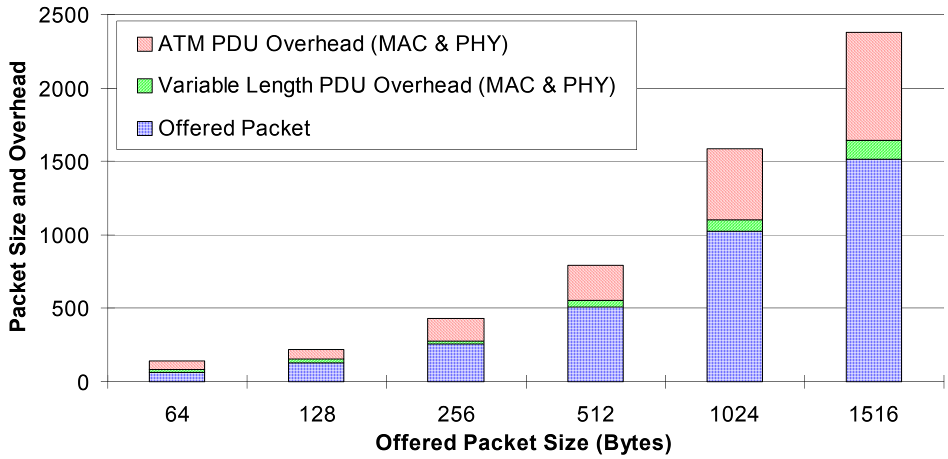

Figure 4-1: Comparison of upstream overheads for carrying 802.3 packets in VL PDUs vs. ATM PDUs.

- ATM cell PDUs: Payload = 48 bytes; OH (PHY + MAC + ATM) = 24 bytes
- Variable Length PDUs: Maximum payload = 256 bytes; Total OH = 20 bytes

Offered Packet (Bytes)	Variable PDU segments including MAC&PHY OH and minislot quantization OH (bytes)	ATM PDU segments including MAC&PHY OH and AAL-5 segmentation OH (bytes)
64	84 (7 minislots)	144 (2*6 = 12 minislots)
128	156 (13 minislots)	216 (3*6 = 18 minislots)
256	276 (23 minislots)	432 (6*6 = 36 minislots)
512	2*276 (2*23 = 46 minislots)	792 (11*6 = 66 minislots)
1024	4*276 (4*23 = 92 minislots)	1584 (22*6 = 132 minislots)
1516	5*276 + 264 (5 *23+22 = 137 minislots)	2304 (32*6=192 minislots)

Table 1: Packet segmentation and overhead comparisons for ATM and VL PDUs (upstream).

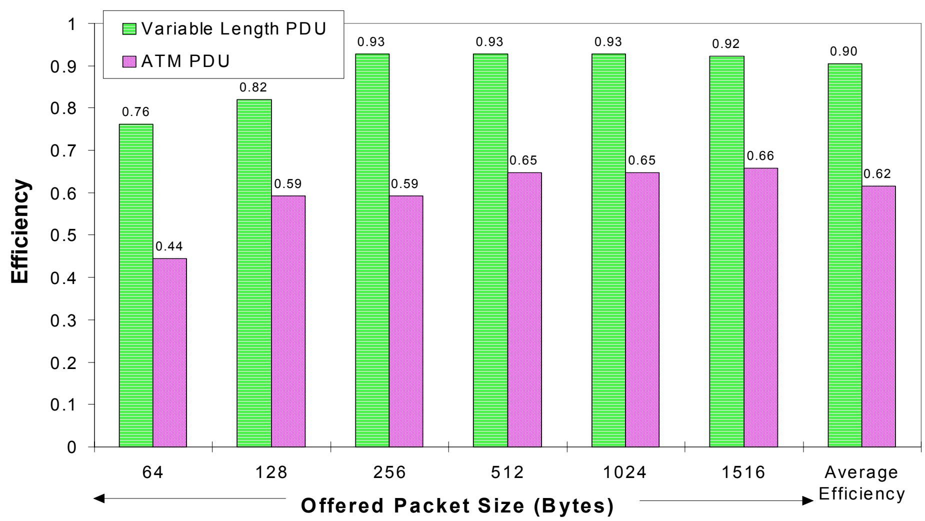

Figure 4-2: Upstream bandwidth efficiencies as a function of packet sizes for ATM and VL PDUs.

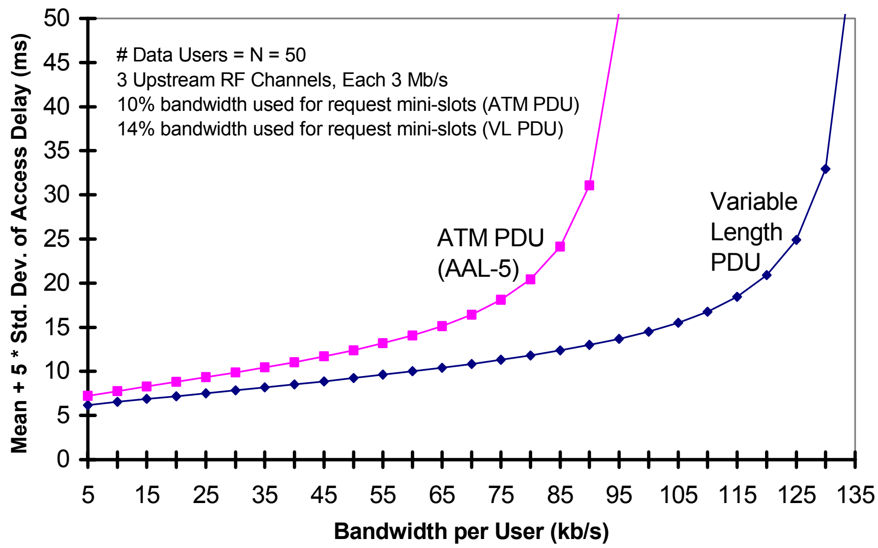

Figure 4-3: Comparison of upstream message access delays for VL vs. ATM PDUs.

5. DOWNSTREAM BANDWIDTH EFFICIENCY COMPARISON FOR ATM VS. VL PDU

In this section, we develop an analytical model for the downstream message delay. The goal is to compare the performance and throughput efficiency of using ATM PDU vs. VL PDU for message transmission over the downstream channel. The MAC protocol is flexible so that the messages (higher layer data units, e.g., IP, 802.3 packets) can be carried in the downstream either in VL PDUs or in ATM PDUs (see Section 2.2). In the latter case, the messages are segmented using AAL-5. The queuing model developed here is generic and applicable to either case. The comparison between ATM and VL PDU usage is best highlighted if the traffic is assumed to be 100% random bursty traffic (see Section 3). In other words, all CMs receive bursty message traffic (e.g., randomly generated IP type of messages as in internet applications). The messages destined for the CMs are queued, and the HC forwards them to the CMs using a FIFO service discipline.

The queuing model developed here is based on an *M/G/1* approximation for message queuing for transmission over a downstream channel. The downstream message access delay, W_D , is defined as the time from arrival of the first byte of a message at the HC to the time the last byte is received at the destination CM.

W_D can be determined as follows by adding the various contributing components:

$$W_D = a_s * (Q_D + T_D) + \tau_p \qquad (5\text{-}1)$$

where,

a_S = average number of PDUs per message (computed from message size distribution and segmentation information),

Q_D = downstream queuing delay at PDU level at the HC,

T_D = average PDU transmission time in the downstream, and

τ_p = one-way propagation time (ms).

Let us define the following parameters for use in derivation of an equation for Q_D:

R = average source rate (same for each receiving CM) (kb/s),

P = average message size without overhead (bytes),

P_S = average PDU size including overhead (bytes),

P_X = average PDU size without overhead (bytes),

$a_S = P/P_X$ = average number of PDUs per message,

λ = average message rate per source (messages/ms),

S_D = average PDU service time,

ρ = downstream channel utilization per RF channel,

C = downstream aggregate channel bandwidth over all RF channels available for data (kb/s),

C_1 = downstream channel bandwidth per RF channel (kb/s),

f_e = fraction of bandwidth used for downstream error protection,

f_c = fraction of bandwidth used for downstream control and management,

β = coefficient of variation of PDU service time (derived from PDU size distribution), and

σ_D = standard deviation of message delay (ms).

The following equations hold:

$$\lambda = \frac{R}{8 * P} \text{ (packets per ms)} \qquad (5\text{-}2)$$

$$S_D = \frac{8 * P_s}{C * (1 - f_e - f_c)} \text{ (ms)} \qquad (5\text{-}3)$$

$$\rho = \lambda * S_D * N = \frac{R * P_s * N}{P * C * (1 - f_e - f_c)} \qquad (5\text{-}4)$$

$$T_D = \frac{8 * P_s}{C_1 * (1 - f_e - f_c)} \quad \text{(ms)} \tag{5-5}$$

Using an *M/G/1* queuing model at the PDU level [19][20], we obtain the following set of equations for performance measures of interest, i.e., Q_D, W_D , and σ_D:

$$Q_D = \frac{\rho * S_D * (1 + \beta^2)}{2 * (1 - \rho)} \quad \text{(ms)} \tag{5-6}$$

$$W_D = a_s * \left(\frac{\rho * S_D * (1 + \beta^2)}{2 * (1 - \rho)} + \frac{8 * P_s}{C_1 * (1 - f_e - f_c)} \right) + \tau_p \quad \text{(ms)} \tag{5-7}$$

$$\sigma_D \approx a_s * Q_D * \sqrt{\frac{4 - \rho}{3 * \rho}} \quad \text{(ms)} \tag{5-8}$$

5.1. *Numerical Examples for Comparison of ATM vs. VL PDU in Downstream*

Messages belonging to random bursty traffic (see Section 3) may also originate from an internet content provider. These messages require transmission in the downstream direction to reach their destination cable modems. The messages can be transported downstream on a cable modem using the VL PDUs or ATM PDUs (with AAL-5 adaptation in the latter case). When data message traffic is subjected to segmentation into smaller ATM PDUs (AAL-5 adaptation), the downstream bandwidth efficiency suffers because of (a) partially filled ATM cells and (b) ATM header overhead. The VL mode may also require some segmentation for meeting the QoS for other (e.g., CBR) traffic that may also be present in the downstream. However, the effect of VL PDU segmentation on the total packet overhead is small compared to that of partially filled ATM cells. For these reasons the overheads for carrying data messages in downstream channels is much lower with VL PDUs as compared to that with ATM PDUs (see Figure 5-1 and Table 2). The width of the VL PDU overhead in Figure 5-1 is almost invisible because it is very small compared to the packet size. The downstream bandwidth efficiencies for ATM and VL PDUs as a function of packet sizes are compared in Figure 5-2. The average bandwidth efficiencies for ATM and VL PDUs (averaged over the packet size distribution described in Section 3) are also shown in Figure 5-2; see the rightmost pair of bars.

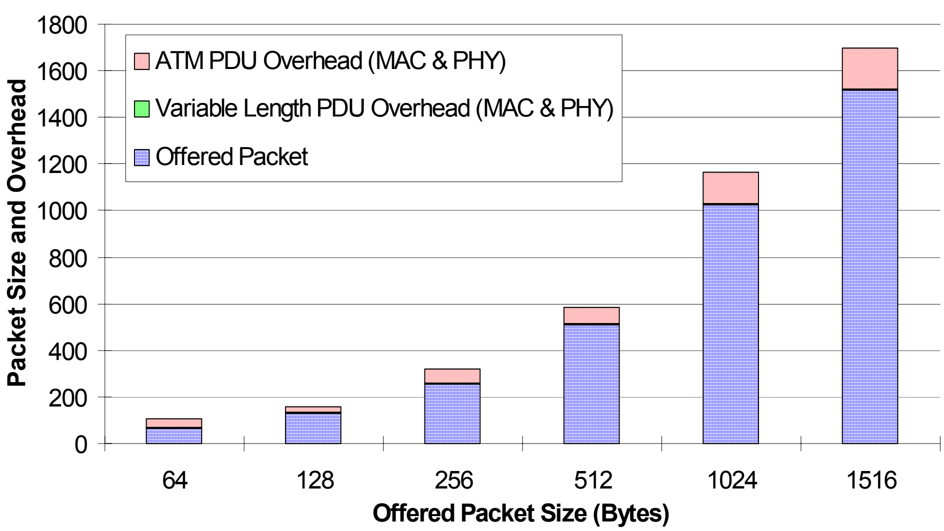

Figure 5-1: Comparison of downstream overheads for carrying messages in VL vs. ATM PDUs.

- ATM cell PDUs: Payload = 48 bytes; OH (MAC + ATM) = 5 bytes
- Variable Length PDUs: No limit on maximum payload; MAC OH = 5 bytes

Offered Packet (bytes)	Variable Length PDUs (bytes)	ATM PDUs (AAL-5) (bytes)
64	69	2 * 53 = 106
128	133	3 * 53 = 159
256	261	6 * 53 = 318
512	571	11 * 53 = 583
1024	1029	21 * 53 = 1113
1516	1521	32 * 53 = 1696

Table 2: Packet segmentation and overhead comparisons for ATM and VL PDUs (downstream).

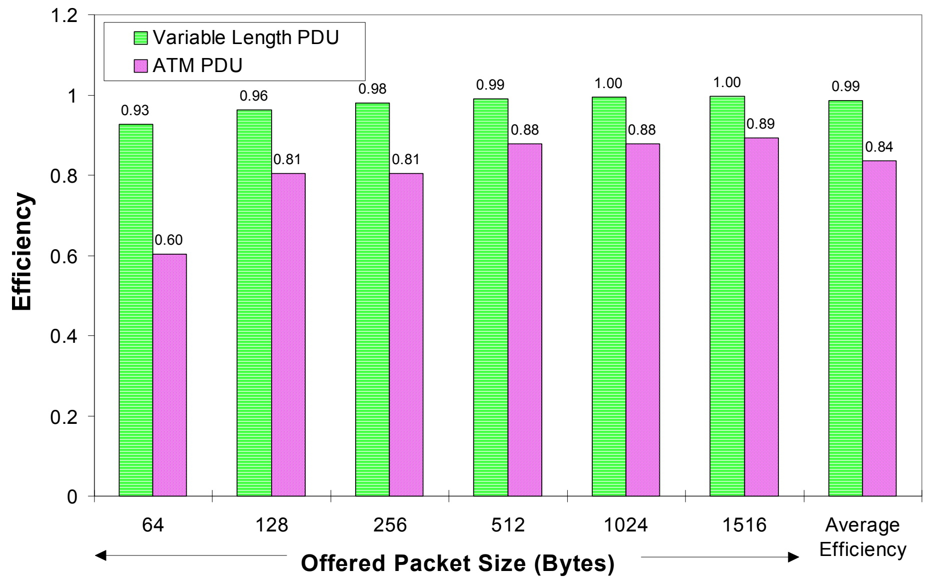

Figure 5-2: Bandwidth efficiencies as a function of packet sizes for ATM and VL PDUs (downstream).

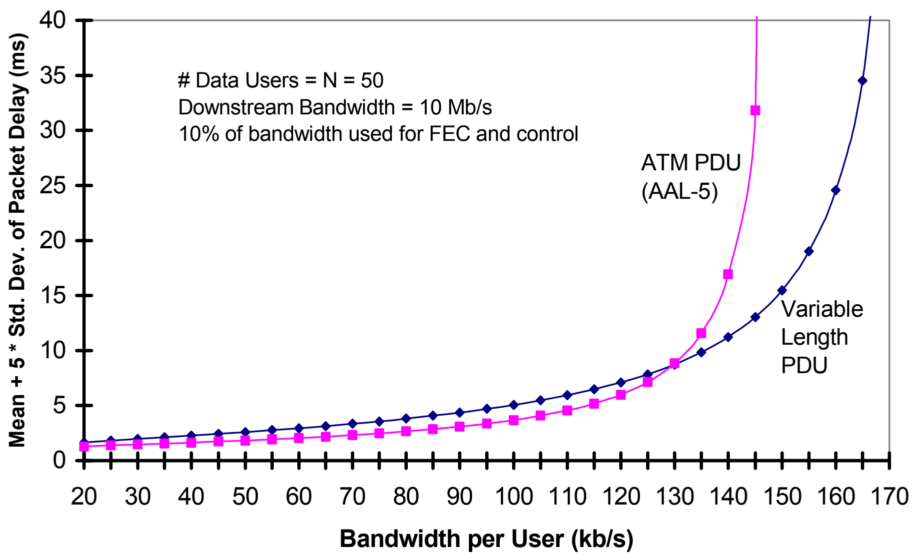

Figure 5-3: Comparison of downstream message access delays for VL vs. ATM PDUs.

The downstream message access delay is defined as the time from arrival of the first byte of a message (or packet) at the HC to the time the last byte (of the same message) is received at the CM. This includes the HC processing and queuing delay, the one-way downstream propagation delay, and the packet transmission delay. The delay plotted in Figure 5-3 is the mean plus five times the standard deviation of message access delay. The downstream bandwidth is assumed to be 10 Mb/s (e.g., using QPSK in one 6 MHz RF channel), and the FEC and control/management overheads are assumed to be 10%. Hence, the ($f_e + f_c$) term in equation (5-7) is equal to 0.1. So the available bandwidth for data traffic is 9 Mb/s. As shown in Figure 5-3, VL PDUs facilitate lower delays and higher throughput. With the use of VL PDUs for the bursty message traffic (described in Section 3), the system saturates at a per-user-throughput value that is approximately 15% higher compared to that for ATM PDUs (see Figure 5-3).

6. UPSTREAM CAPACITY ENHANCEMENT DUE TO DYNAMIC REQUEST MINISLOTS

In this section, we describe the operation of MAC protocols for three cases with regard to the use of minislots in the upstream channel. It is assumed that the information packets are all of a fixed size (e.g., ATM cells) for the purpose of this study. For the remainder of this paper, we use the term "data packet" instead of "information packet" for ease, with understanding that "data" may mean voice, computer-data or video information.

6.1. *MAC Protocol Without Minislots*

Figure 6-1 shows an upstream frame for a MAC protocol that does not use minislots. In this scheme, when a batch or burst of data packets (or ATM cells) arrive at a CM, the first packet of the burst is transmitted in a contention mode using a contention time slot (denoted by C). A request for bandwidth for the remainder of the packets of that burst is transmitted along with the first packet (also known as piggy-backing of request). When the request is granted by the HC, the subsequent packets of the burst are transmitted in a reservation mode, in time slots denoted by R in Figure 6-1, as per allocation by the HC. In the case of a reservation-oriented transaction, a connection set-up request is made by sending a request message in one of the C slots. Subsequently, the call is admitted, depending on bandwidth availability, and reserved *(R)* slots are periodically allocated to the call for its duration. The frame duration in Figure 6-1 is shown to be 2 ms as an example. The round-trip propagation delay accounts for about 1 ms, and about 1 ms is budgeted for transmission and HC/CM processing times. At an upstream channel rate of 2.56 Mb/s, typically there would be about 10 data (ATM cell) time slots in a 2 ms frame duration. As shown in Figure 6-1, at least one time slot in each frame is designated for contention access so that new call requests and other signaling messages may be transmitted without undue delay.

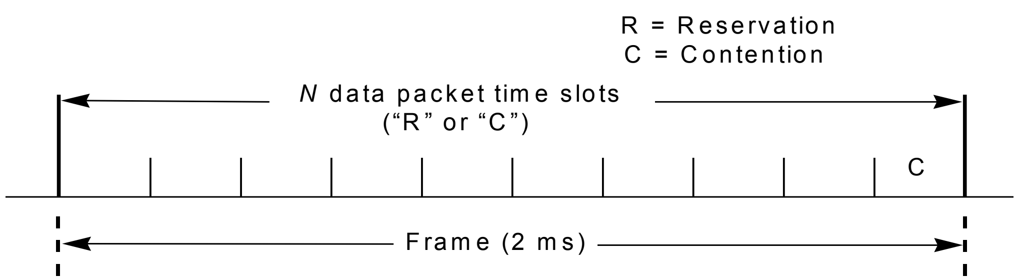

Figure 6-1: Upstream frame without minislots.

6.2. *MAC Protocol With Fixed Number of Minislots*

Figure 6-2 illustrates an upstream frame for a MAC protocol that operates with a fixed number of minislots. Let N represent the total number of data (packet) time slots per frame. The value of N is a function of the frame duration, the physical link bandwidth, and the packet size. Out of a total of N data time slots in a frame, the duration (or span) of j_f data slots is designated by the HC for use as minislots.

Thus the number of minislots is $k * j_f$ per frame. It is assumed here that a data time slot is the same size (i.e., duration) as k contention request minislots (k = 4 in the example of Figure 6-2). The value of k depends on the physical layer overheads (guardtime, preamble, FEC, etc.) and the length of the request message transmitted in a minislot. For an upstream transmission rate of 2.56 Mb/s, typically 4 to 6 minislots can be accommodated in the time duration of one data (ATM cell) time slot. In Figure 6-2, n_1 is the number of time slots in use by the reservation-oriented traffic, and hence typically the value of n_1 remains constant for a long duration. The number of time slots available for contention-oriented bursty data packets is n_2. The value of j_f is determined as function of N and k only, and is not adjusted when n_1 changes [18]. Hence, we refer to this scheme as a MAC protocol with fixed number of minislots.

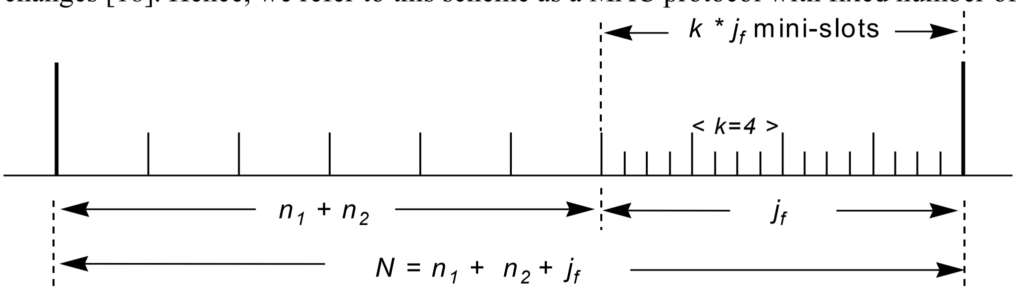

Figure 6-2: Upstream frame with a fixed number of minislots.

6.3. *MAC Protocol With Variable Number of Minislots*

Figure 6-3 illustrates an upstream frame for a MAC protocol that uses a variable number, j_v , of minislots. In this scheme, the number of request minislots , j_v , is a function of N , k , and n_1 and hence varies with the number of packet time slots, n_1 , used by reservation-oriented traffic. Hence, j_v may dynamically vary from one frame to the next as a function of the bandwidth currently in use by the reservation-oriented traffic. The potential benefits of this scheme may be qualitatively explained as follows (the quantitative comparisons will be provided in subsequent sections). When a greater number of packet time slots are being used in a reservation mode for a relatively long period of time, fewer minislots are needed for contention requests. Just enough minislots should be created so that average throughput (per frame) of the contention request minislots closely matches the number of available data time slots (n_2 per frame) for the contention-oriented bursty messages. An algorithm for varying the number of minislots that is based on maintaining this dynamic balance will be described later in this paper.

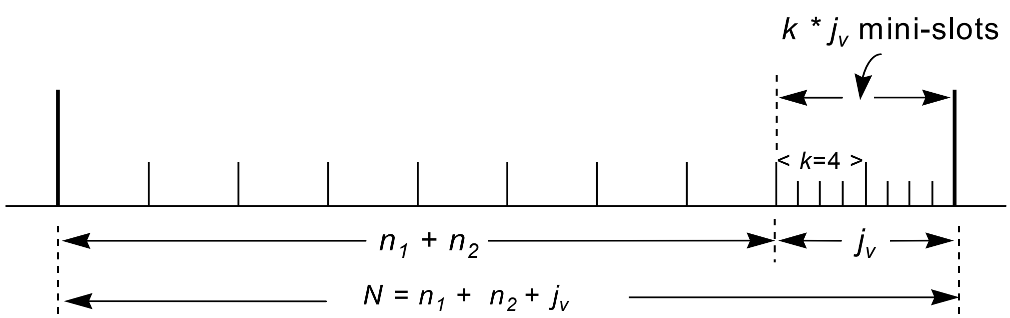

Figure 6-3: Upstream frame with a variable number of minislots.

6.4. *Throughput Efficiency Analysis*

Let us define the following parameters to assist in the analytical derivations:

α = physical layer (PHY) efficiency (including effects of guardtime, preamble, FEC, etc.),

N = total number of packet time slots per frame (including those used for creation of minislots),

k = number of minislots that can be accommodated within the duration of a packet time slot (k depends on α and quantization effects thereof),

n_1 = number of packet time slots in use by reservation-oriented traffic (long holding time),

n_2 = number of packet time slots available for bursty messages

$$= \begin{cases} N - n_1 & \text{for protocol with no mini-slots} \\ N - n_1 - j_f & \text{for protocol with fixed number of mini-slots} \\ N - n_1 - j_v & \text{for protocol with variable number of mini-slots} \end{cases}$$

E_a = throughput efficiency of all traffic for the MAC protocol without minislots,

E_f = throughput efficiency of all traffic for the MAC protocol with fixed number of minislots,

E_v = throughput efficiency of all traffic for the MAC protocol with variable number of minislots,

S_a = throughput efficiency of bursty messages for the MAC protocol without minislots,

S_f = throughput efficiency of bursty messages for the MAC protocol with fixed number of minislots,

S_v = throughput efficiency of bursty messages for the MAC protocol with variable number of minislots,

u_1 = throughput of reservation-oriented traffic (%) = $\dfrac{100 * n_1 * \alpha}{N}$,

B = average bursty message length including segmentation and PHY/MAC overheads (bytes),

B_a = size of a packet (one data PDU) including PHY/MAC overheads (bytes),

$\ell = \dfrac{B}{B_a}$ = average number of packets per bursty message.

6.4.1. *Determination of Number of Minislots*

In this section, we present the methodology for determining the number of request minislots. The request messages are subject to collisions. A suitable Collision Resolution Algorithm (CRA) is used to facilitate successful retransmission (see Section 1.2) in the event of collision(s). With the use of a simple CRA such as the random binary exponential back-off algorithm, the theoretical throughput efficiency of a minislot is 37%. For practical reasons, primarily based on delay considerations, the efficiency of each minislot is would be closer to 0.9 * 37% = 33.3 % [2][9]. So three request minislots are needed to achieve 100% throughput efficiency in an associated data time slot (for random bursty messages of size one packet each).

MAC With Variable Number of Request Minislots:

Based on the preceding observations, the ratio of the number of minislots to the number of data slots should be at most 3:1. This ratio can be smaller when the bursty messages are multi-packet messages. When a bursty message consists of multiple packets, only the first packet generates a contention request, and when that request successfully reaches the Head-end Controller (HC), the rest of the packets of the multi-packet messages are transmitted based on reservations allocated by the HC. Given that the average bursty message length is ℓ, on average ℓ data packet time slots are utilized for each request message that successfully makes it through the minislot contention process. However, the bursty messages are randomly varying in size, and are often smaller than ℓ. Hence, we introduce a multiplication factor, f_m (f_m < 1), for conservative use of the benefit of ℓ. We compute an effective message length, ℓ_e, (in number of packets or MAC PDUs) as follows:

$$\ell_e = max\left\{\lfloor \ell * f_m \rfloor, 1\right\} \tag{6-1}$$

We set f_m = 0.5 for the numerical examples so that when $\ell \geq 4$, ℓ_e helps to reduce the number of minislots needed per frame.

Further, in computing the number of minislots, it is recognized that even one minislot per frame is sufficient for the request messages originating from reservation-oriented traffic. Such requests are very sparse compared to those originating from bursty messages because one reservation is made per reservation-based transaction which lasts a long time. At least one data time slot is always designated by the HC for use as k minislots. Hence, it is safe to not require any additional minislots to be created for the reservation-oriented traffic.

Algorithm For Computing The Variable Number Of Minislots (j_v):

Based on the insights presented so far into the nature of the relationship between minislot throughput and data slot throughput, the following equations represent a procedure for computing the number, j_v , of packet time slots that are designated by the HC to be used as $k* j_v$ minislots for the variable minislot case. These equations help optimize the throughput for the bursty message traffic as a function of n_1 , N, k, and ℓ_e :

$$\text{Let} \quad j_x = max\left\{\left\lceil \frac{3(N-n_1)}{3+k*\ell_e} \right\rceil, 1\right\}, \qquad \text{[A]}$$

$$\text{and} \quad \mathrm{P} = Logical \quad \{j_x \geq 2 \; and \; 0.33*(j_x - 1)*k*\ell_e \geq N - n_1 - j_x\}, \qquad \text{[B]} \tag{6-2}$$

then,

$$j_v = \begin{cases} j_x & \text{if P = FALSE,} \\ j_x - 1 & \text{if P = TRUE} \end{cases} \qquad \text{[C]}$$

In equation [6-2A] above, an initial value for j_v is computed based on the observations made earlier. This equation is further refined in equations [6-2B] and [6-2C], by noting that if j_x is at least 2 and the throughput (in equivalent packet units) of requests per frame for bursty messages via (j_x - 1) * k contention minislots exceeds the number of data time slots, N - n_1 - j_x , available for bursty messages (for the case of j_x * k minislots), then the value computed in equation [6-2A] is recognized to be a little too high. Accordingly, an adjustment is made in [6-2C] by decrementing the value obtained in [6-2A] by one to arrive at the final value for j_v. If this adjustment step were not followed, then it is possible that the request throughput per frame frequently exceeds the number of available data time slots for the bursty messages. Thus the data queue for bursty messages could back up and overflow in the next several upstream frames.

Figure 6-4 illustrates that for n_1 = 5, N = 10, k = 4 and ℓ_e = 1, we would initially compute a value 3 for j_x following equation [6-2A]. However, it is recognized that for the value of j_v = 3, the minislot request throughput is 3.96 while the maximum available bursty message packet throughput is smaller at n_2 = 2 packets per frame. However, for j_v = 2, the maximum available bursty message packet throughput is 3 while the request throughput is 2.64. The actual packet throughput for the bursty message traffic would be the smaller of these two quantities. Thus with j_v = 2, the throughput in consideration is min(3, 2.64) = 2.64 packets per frame (see the lower part of Figure 6-4), which is better than that with j_v = 3, since min(2, 3.96) = 2 (see the upper part of Figure 6-4).

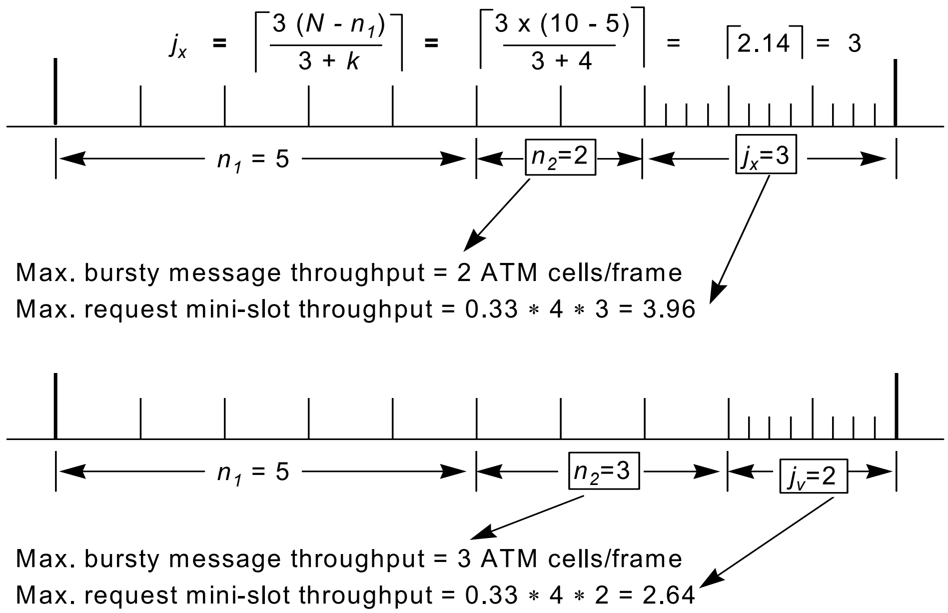

Figure 6-4: Illustration of how the refinement in the value of j_v using equations [2B] and [2C] helps in fine tuning the throughput efficiency (for $n_1 = 5$, $N = 10$, $k = 4$, and $\ell_e = 1$).

It may also be noted that if the distribution of bursty message sizes is not known a priori but estimated by traffic measurement during system operation, then the algorithm of equation (6-2) can still be effectively used by incorporating the most recent measurement estimate of ℓ_e. In doing so, the system performance can be dynamically tuned to the changing nature of bursty message traffic characteristics over time.

MAC With Fixed Number of Request Minislots:

The following somewhat simplistic rationale may be followed to approximately estimate the number of minislots per frame for a MAC with a fixed number of minislots. Three request minislots per data time slot would be required if the traffic were 100% purely sporadic traffic with single packet messages ($\ell = 1$). However, some of the traffic is reservation-oriented with long transaction time and some of it is bursty messages with a mix of message lengths. Therefore, it may be reasoned that assigning 2 minislots per data slot would be sufficient [18]. By doing so the overhead due to the presence of minislots is better compared to the case of 3:1 ratio. Hence, the following equation ensues for the number, j_f, of packet time slots that are designated for use as $k^* j_f$ minislots (per frame):

$$j_f = max \left\{ \left\lceil \frac{2N}{2+k} \right\rceil , 1 \right\} . \tag{6-3}$$

6.4.2. *Throughput Efficiency Results*

MAC With No Request Minislots:

In this case, the MAC operates essentially like CSMA/CD as far as throughput goes. Due to the presence of multi-packet bursty messages, the throughput efficiency is better than that of slotted Aloha (with random binary exponential back-off algorithm for collision resolution). The average efficiency, β, for the contention time slots allocated to bursty messages is given:

$$\beta = \sum_{i} p_i * \eta_i \, , \tag{6-4}$$

where p_i are the probabilities describing the frequency of various message lengths (see Section 2.1) and η_i are the contention throughput efficiencies (similar to those for the CSMA/CD scheme) corresponding to different message lengths (see [2] for details, also [22]-[24]). The message lengths shown in Section 2.1 are (2, 3, 6, 11, 22, 32) packets (ATM cells) with probability distribution given by (0.6, 0.06, 0.04, 0.02, 0.25, 0.03). The corresponding values of η_i for these message lengths obtained from the analysis in [2] are approximately (0.4, 0.45, 0.55, 0.65, 0.7, 0.8). The value of β now computes to about 0.5 or 50%. Taking the PHY efficiency, α, into account, the throughput efficiency, S_a, of bursty messages for the MAC protocol without minislots is as follows:

$$S_a = \frac{(N - n_1) * \beta * \alpha}{N} \tag{6-5}$$

Further, the throughput efficiency, E_a, of the combined (reservation-oriented plus random bursty message) traffic for the MAC protocol without minislots is:

$$E_a = \frac{n_1 * \alpha}{N} + S_a = \frac{[n_1 + (N - n_1) * \beta] \alpha}{N} \tag{6-6}$$

MAC With Fixed Number of Request Minislots:

The throughput efficiency, S_f, of the random bursty messages for the MAC protocol with fixed number of minislots is as follows:

$$S_f = \frac{\min\{(N - n_1 - j_f), 0.33 * k * j_f * \ell_e\} \alpha}{N} \tag{6-7}$$

The throughput efficiency, E_f, of the combined (reservation-oriented plus random bursty message) traffic for MAC protocol with fixed number of minislots is:

$$E_f = \frac{n_1 * \alpha}{N} + S_f = \frac{[n_1 + \min\{(N - n_1 - j_f), 0.33 * k * j_f * \ell_e\}] \alpha}{N} \tag{6-8}$$

MAC With Variable Number of Request Minislots:

The throughput efficiency, S_v, of the random bursty messages for the MAC protocol with variable number of minislots is as follows:

$$S_v = \frac{\min\{(N - n_1 - j_v), 0.33 * k * j_v * \ell_e\} \alpha}{N} \tag{6-9}$$

The throughput efficiency, E_v, of the combined (reservation-oriented plus random bursty message) traffic for the MAC protocol with variable number of minislots is:

$$E_v = \frac{n_1 * \alpha}{N} + S_v = \frac{[n_1 + \min\{(N - n_1 - j_v), 0.33 * k * j_v * \ell_e\}] \alpha}{N} \tag{6-10}$$

6.5. *Numerical Examples*

The parameter values used for generating numerical results that are shown in Figure 6-5, Figure 6-6, and Figure 6-7 are as follow: N = 10, α = 0.8, k = 5, β = 0.5, ℓ = 8, f_m = 0.5, and $\ell_e = f_m * \ell = 4$. The numbers shown in Figure 6-5, namely j_f and j_v, should be multiplied by k to get the number of minislots used per frame for the cases of MAC with fixed number and MAC with variable number of minislots, respectively. The values of j_v and j_f shown in Figure 6-5 are computed in accordance with equations (6-2) and (6-3), respectively. As the reservation-oriented traffic increases, the MAC with variable minislots dynamically varies the number of minislots so as to maximize the throughput for the bursty message traffic in the time slots that are available for the latter traffic.

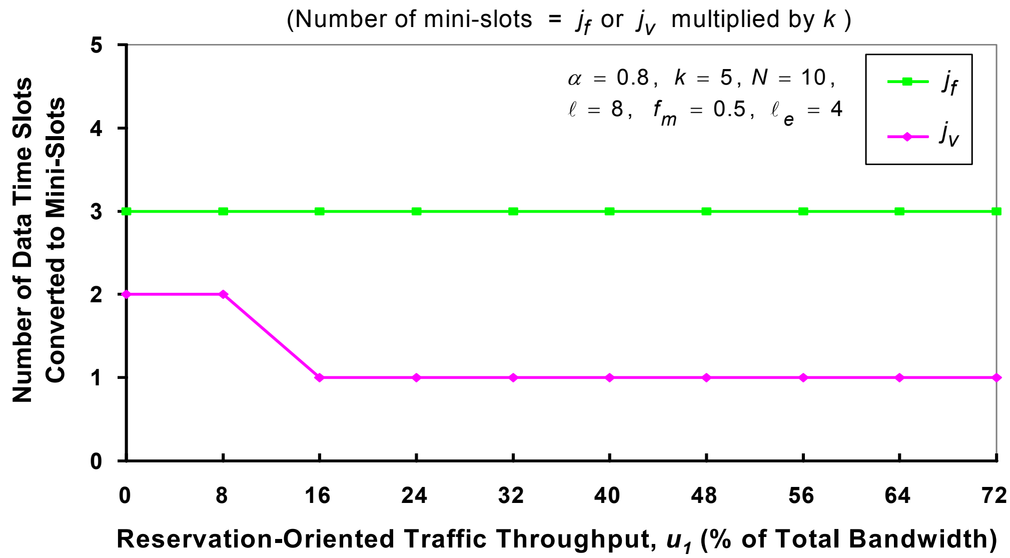

Figure 6-5: The number of packet time slots converted to request minislots as a function of the throughput used by reservation-oriented traffic.

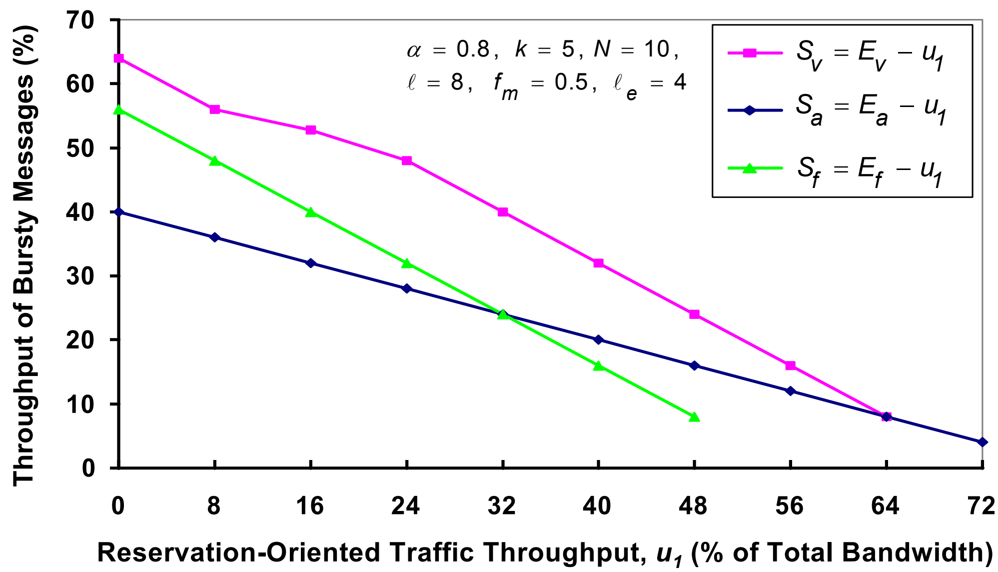

Figure 6-6: Throughput efficiency of random bursty traffic as a function of the throughput used by reservation-oriented traffic.

Figure 6-6 and Figure 6-7 show the throughput efficiencies for the bursty messages and combined traffic, respectively. Clearly, the MAC protocol with variable minislots provides the best throughput efficiency because it dynamically adjusts the number ($= j_v * k$) and hence the overhead due the presence of minislots as a function of the given mix of traffic. The plots for the case of MAC protocol with a fixed minislots (denoted by S_f in Figure 6-6 and E_f in Figure 6-7) do not extend all the way to the right because 3 time slots out of 10 are always given over to creating minislots. Accordingly, the number of time slots that can be allocated to reservation-oriented traffic is limited to 7 (out of 10). In the case of a MAC protocol with variable number of minislots, it is possible to lower the value of j_v all the way to one for cases when there is high amount of reservation-oriented traffic in the system. Hence, the plots for the case of variable minislots (denoted by S_v in Figure 6-6 and E_v in Figure 6-7), extend farther out to higher values of reservation-oriented traffic throughput.

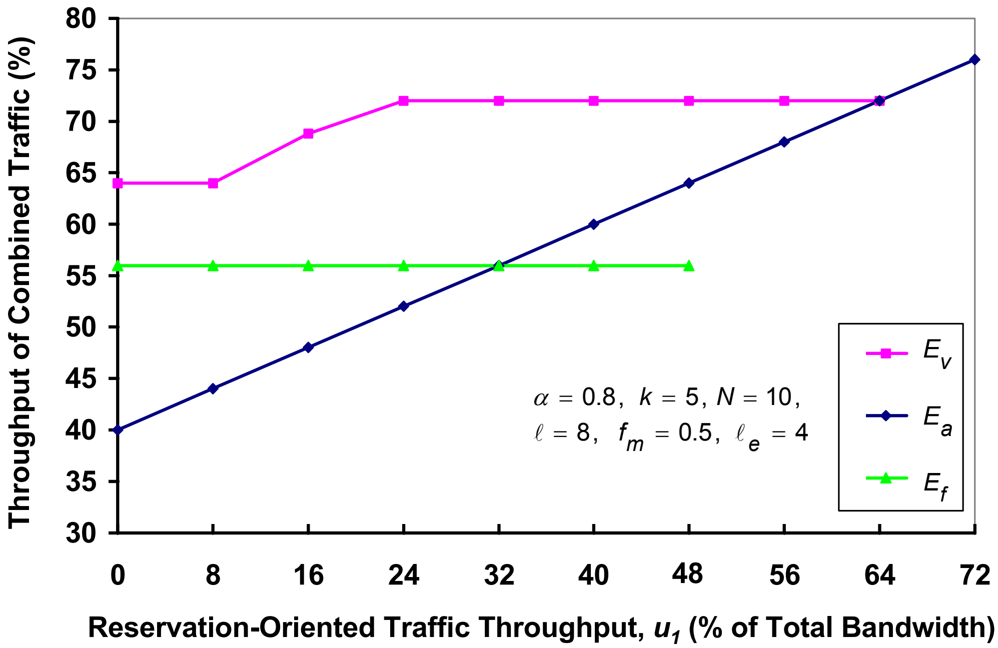

Figure 6-7: Throughput efficiency of the combined (reservation-oriented plus random bursty) traffic as a function of the throughput used by reservation-oriented traffic.

7. DELAY AND EFFICIENCY COMPARISONS FOR ALTERNATIVE METHODS OF SUPPORTING STM IN DOWNSTREAM

In Section 2.3, three methods were described for supporting DS0 traffic in the downstream for telephony applications. Let n denote the number of voice calls (64 kb/s DS0 connections) that are currently in progress. Let V_a , V_b , and V_c denote the downstream bandwidths for supporting n voice calls for the three methods (a), (b), and (c). Including the effects of signaling and ATM overheads as described in Section 2.3, the following equations hold for V_a , V_b , and V_c .

(a) Using an STM byte region in 125 μs subframes:

$$V_a = \left(\left\lceil \frac{n}{16} \right\rceil + n \right) * 64 \ kbps \tag{7-1}$$

This method is the usual strict STM approach. The fixed downstream voice delay for this case is 125 μs.

(b) Scheduling composite ATM cell every 125 μs:

$$V_b = \left\lceil \frac{n}{45} \right\rceil * \left(\frac{53 * 8}{0.125} \right) kbps \tag{7-2}$$

The voice delay for this case is 250 μs including the receiver buffer play-out delay for smoothing over the limited delay jitter that may happen due somewhat asynchronous nature of ATM CBR service.

(c) Scheduling composite ATM cell every 1 ms:

$$V_c = \left\lceil \frac{n}{5} \right\rceil * (53 * 8) \ kbps \tag{7-3}$$

The voice delay for this case is 1.25 ms including the receiver buffer play-out delay for smoothing over the limited delay jitter that may happen to due somewhat asynchronous nature of ATM CBR service.

7.1. *Numerical Results on Bandwidth Efficiencies for Telephony in Downstream*

The comparison of bandwidth needed for the three approaches of supporting telephony in the downstream is shown in Figure 7-1. Method (a) has the least delay and most bandwidth efficiency. However, one may use composite ATM cells as in methods (b) and (c) to support DS0 connections while avoiding a synchronous 125 μs framing structure in the downstream. If such is the case, it is evident from Figure 7-1 that method (b) is very bandwidth inefficient when the voice call volume is low. If about 2 ms (1.25 ms due to buffering plus the rest for propagation delay) of the available access delay budget for voice can be allocated for the downstream channel, then method (c) would be desirable over (b). If echo cancellors are used in the access loop, then the delay budget of 2 ms can be easily allocated to downstream.

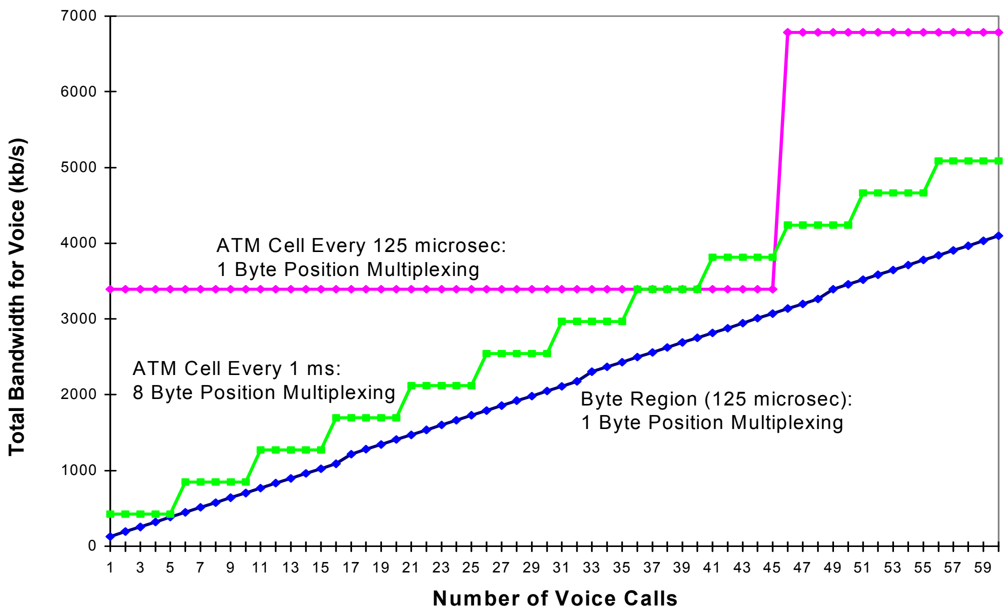

Figure 7-1: Comparison of bandwidth efficiencies for alternative approaches for supporting DS0 connections for telephony application in the downstream.

8. CONCLUSIONS

Due to the wide-ranging traffic characteristics and performance requirements of applications that are supported over broadband HFC and wireless access networks, it is important to design MAC protocols with several key features for flexible and efficient operation. In this paper, we described several key features that enable superior performance and efficiency in Medium Access Control (MAC) protocols. These features include: (1) minislots for bandwidth allocation granularity, (2) flexible transport capability for fixed-length (ATM) protocol data units (PDUs) and variable-length (VL) PDUs (e.g., IP, 802.3 frames, IPv6), (3) means for low delay transport of STM voice information, (4) dynamically variable minislots for contention request resolution, and (5) piggy-backing for additional transmission requests. We also included in this study a comparison of some alternative MAC protocol features.

For upstream channels, the use of variable length (VL) PDUs for carrying data traffic results in about 50% higher bandwidth efficiency as compared to the use of ATM PDUs (with adaptation based on AAL-5). The corresponding number for the increase in bandwidth efficiency for downstream channels is 10 to 15%. Accommodation of VL PDUs as well as ATM PDUs is best accomplished in the upstream by means of the minislot granularity. The support of VL PDUs in the downstream may be accomplished by creating two separate regions (or byte-streams) for data transmission, for carrying ATM and VL PDUs, respectively. The boundary between these two data regions should be dynamic and specified by the HC. A system designed on the above principles would initially carry mostly VL PDUs to efficiently support 10baseT interfaces at the customer premises, and it can evolve gracefully to ATM PDU based transport in the access network as the information appliances and backbone networks become increasingly ATM capable. We also discussed and compared alternatives for supporting low-delay requiring DS0 (and n*DS0) connections, with the telephony application in mind.

We compared MAC protocols for three cases with regard to request minislots: (1) with no minislots (in this case, the first of a batch of data packets from a station is transmitted in a full data slot using contention mode access, and also carries with it a reservation request for the remainder of packets in that batch), (2) with fixed number of minislots per frame, and (3) with dynamically variable number of minislots per frame. We also described an algorithm for dynamically varying number of minislots as a function of the traffic mix. The results show that a MAC protocol with dynamically variable minislots has the highest throughput efficiency amongst the different alternatives discussed.

The IEEE 802.14 standards specification (draft) [14] and the MCNS (a cable industry consortium) specification [15][16] on cable modems include several of the key features discussed in this paper as part of their MAC protocols.

Acknowledgements:

The author would like to thank Chia-Chang Li and James Dail of AT&T Laboratories, and Bharat Doshi, Subrahmanyam Dravida, Peter Magill, Miles Murdocca, and Curtis Siller, Jr. of Lucent Technologies - Bell Laboratories for comments and suggestions during the course of this work.

REFERENCES

1. K. Sriram, C. Li, P. Magill, N.A. Whitaker, J.E. Dail, M.A. Dajer, C.A. Siller, Jr., "An Adaptive MAC-Layer Protocol for Multi-Service Digital Access via Tree and Branch Communication Networks," *SPIE Proceedings: Hybrid Fiber-Coax Systems*, vol. 2609, pp. 10-21, October 1995.

2. J.E. Dail, M.A. Dajer, C.-C. Li, P.D. Magill, C.A. Siller, Jr., K. Sriram, and N.A. Whitaker, "Adaptive Digital Access Protocol: A MAC Protocol for Multiservice Broadband Access Networks," *IEEE Communications Magazine*, pp. 104-112, March 1996.

3. B.T. Doshi, S. Dravida, P.D. Magill, and C. A. Siller, Jr., and K. Sriram, "A Broadband Multiple Access Protocol for STM, ATM, and Variable Length Data Services on Hybrid Fiber-Coax Networks," *Bell Labs Technical Journal*, Volume 1, Number 1, Summer 1996, pp. 36-65.

4. K. Sriram and P.D. Magill, "Enhanced Throughput Efficiency by Use of Dynamically Variable Request Minislots in MAC Protocols for HFC and Wireless Access Networks," *Telecommunications Systems* (special issue on Multimedia), 1998 (to appear).

5. H. Xie, P. Narasimhan, R. Yuan, and D. Raychaudhuri, "Data Link Control for Wireless ATM Access Channels," *Proceedings of the IEEE ICUPC* (1995), pp.753-757.

6. M. Karol, Z. Liu, and P. Pancha, "Implications of Physical Layer Overhead on the Design of Multiaccess Protocols,'' *IEE Electronics Letters*, vol. 32, pp. 2062-2063, October 24, 1996.

7. M. Karol, Z. Liu, and P. Pancha, "Performance Implications of Physical Layer Overhead on Demand-Assignment MAC Protocols,'' *Proc. of the Eleventh Annual IEEE Workshop on Computer Communications*, Reston, VA, Sept. 1996.

8. D. Sala, J.O. Limb, and S. Khaunte, " Adaptive MAC protocol for a cable modem," Georgia Inst. Tech. Technical Report, GIT-CC-97/14, May 1997.

9. N. Golmie, S. Masson, G. Pieris, and D.H. Su, "A MAC protocol for HFC networks: Design issues and performance evaluation," *Computer Communications, vol. 20, no. 12, pp. 1042-1050,* November 1997.

10. L. F. Merakos and C. Bisdikian, "Delay analysis of the n-ary stack random access algorithm," *IEEE Trans. on Information Theory*, pp. 931-942, Sept. 1988.

11. C. Bisdikian, B. McNeil, R. Norman, "ms-START: A random access algorithm for the IEEE 802.14 HFC network," *Computer Communications*, (special issue on "Recent advances in networking technology"), pp. 876-887, Sept. 1996.

12. C. Bisdikian, "The n-ary stack algorithm for the wireless random access channel," *Mobile Networks and Applications*, vol. 2, pp. 89-99, 1997.

13. R. Citta, J. Xia, and D. Lin, "A Tree-Based Algorithm with Soft Blocking," IEEE Project 802.14, Doc. No. IEEE 802.14/96-244, November 4, 1996.

14. IEEE 802.14 Working Group, web page at http://www.walkingdog.com

15. MCNS Holdings, Inc., web page at http://www.cablemodem.com

16. CableLabs, Boulder, Colorado, web page at http://www.cablelabs.com/cable-spec.html

17. R. Gusella, "A Measurement Study of Diskless Workstation Traffic on an Ethernet", *IEEE Transactions on Communications*, vol. 38, no. 9, pp. 1557-1568, September 1990.

18. C.-T. Wu and G. Campbell, "CBR channels on DQRAP-based HFC network," *SPIE Proceedings: Hybrid Fiber-Coax Systems*, vol. 2609, pp. 154-167, October 1995.

19. K. Sriram and W. Whitt, "Characterizing Superposition Arrival Processes in Packet Multiplexers for Voice and Data," *IEEE Journal on Selected Areas in Commun.*, vol. SAC-4, no. 6, pp. 833-846, September 1986.

20. K. Sriram, "Methodologies for bandwidth allocation, transmission scheduling, and congestion avoidance in broadband ATM networks," *Proc. of Globecom'92*, Orlando, pp. 1545-1551, December 1992. Also, in the *Computer Networks and ISDN Systems*, special issue on "Traffic Issues in ATM Networks", pp. 43-59, vol. 26, September 1993.

21. J.E. Dail, C.-C. Li, P.D. Magill, K. Sriram, and N.A. Whitaker, "Method and Apparatus Enabling Synchronous Transfer Mode and Packet Mode Access for Multiple Services on a Broadband Communication Network," U.S. Patent No. 5,570,355, October 29, 1996.

22. "Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications", ANSI/IEEE Std. 802.3-1985.

23. L.G. Roberts, "ALOHA Packet System With and Without Slots and Capture", *Computer Communications Review* 5(2), April 1975, pp.28-42.

24. S.S. Lam, "Packet Broadcast Networks - A Performance Analysis of the R-ALOHA Protocol", *IEEE Transactions on Computers*, Vol. C- 29, No. 7, pp. 596-603, July 1980.